

# GPLSI-UH LETO v1.0: Motor de aprendizaje a través de ontologías

## GPLSI-UH LETO v1.0: Learning Engine Through Ontologies

**Resumen de la aplicación:** LETO es un marco de aprendizaje de ontologías diseñado para extraer conocimiento de una variedad de fuentes. Estas fuentes pudieran ser datos estructurados y no estructurados, y de ellas se podrá descubrir, actualizar continuamente, enriquecer e integrar información relevante como parte de un único conocimiento semántico. En la actual versión 1.0 se limita a la extracción de conocimiento desde datos no estructurados, i.e. textos en lenguaje natural, siguiendo el modelo semántico publicado en [EGM2018]. Entre sus funcionalidades está la extracción de entidades y relaciones semánticas desde fuentes textuales; la transformación de esta información en elementos interrelacionados mediante técnicas de agrupamientos; y finalmente generación de ontologías representativas del contenido procesado. Se proporciona un punto de acceso API, y una herramienta visual para la manipulación de procesos y visualización de las ontologías obtenidas [EMA2019].

**Summary:** LETO is an ontology learning framework designed to extract knowledge from a variety of sources. These sources may be structured and/or unstructured data, and from them we can discover, continuously update, enrich and integrate relevant information as part of a single semantic knowledge resource. The current 1.0 version is limited to the extraction of knowledge from unstructured data, i.e. natural language texts, following the semantic model published in [EGM2018]. Among this version's functionalities are the extraction of entities and semantic relations from textual sources; the transformation of such information into linked elements through clustering techniques; and finally, the generation of representative ontologies of the processed content. An API access point as well as a visual tool for the manipulation of processes and visualization of the obtained ontologies is provided [EMA2019].

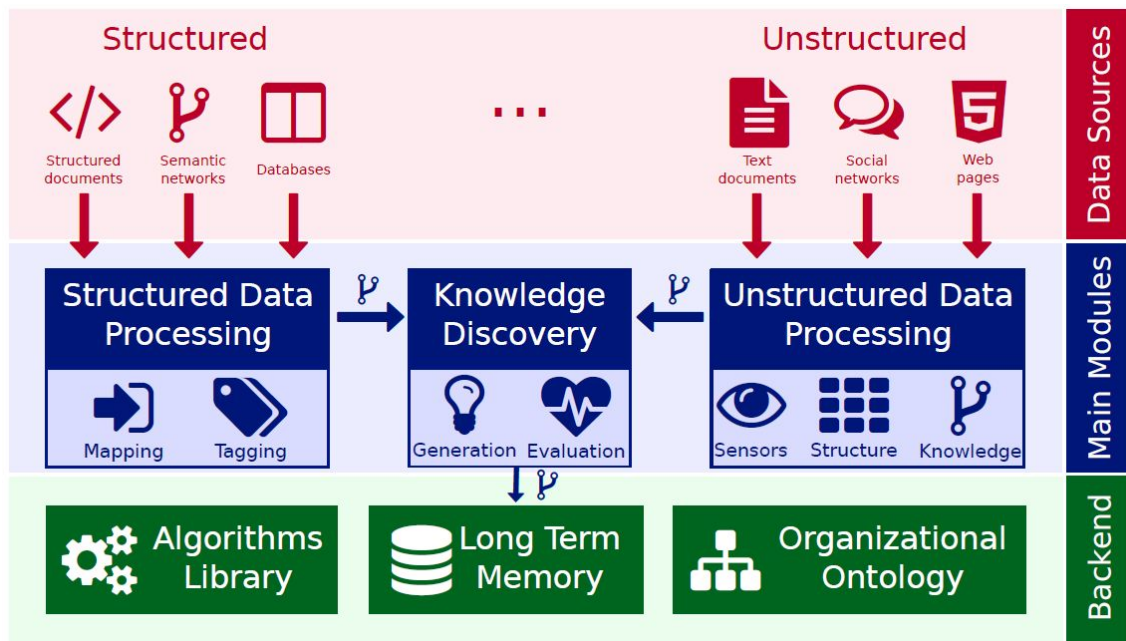


Fig. Arquitectura [EMA2019].

## Especificaciones técnicas/ Technical Specifications

Lenguaje de programación/ Development Language: Python

Entorno Operativo/ Operating Environment: Linux/ Windows

Versión/Version: 1.0

Punto de Acceso REST: `localhost:3000/API`

Estructura de ficheros/ File structure:

- **leto**
  - data
  - docker-compose.base.yml
  - docker-compose.dev.yml
  - docker-compose.prod.yml
  - Dockerfile
  - download\_corpora.sh
  - leto
  - leto.sh
  - leto-ui
  - notebooks
  - pipelines
  - Readme.md
  - requirements.txt
  - scripts
- **leto/leto**
  - \_\_init\_\_.py
  - knowledge
  - storage
  - structured
  - unstructured
- **leto/leto/knowledge**

- evaluation.py
- generation.py
- \_\_init\_\_.py
- rules.py
- **leto/leto/structured**
  - \_\_init\_\_.py
  - mapping.py
  - tagging.py
- **leto/leto/unstructured**
  - \_\_init\_\_.py
  - knowledge
  - sensorial
  - structural
- **leto/leto/unstructured/knowledge**
  - actions.py
  - entity\_emotion\_ontology.py
  - \_\_init\_\_.py
  - run\_cluster.py
- **leto/leto/unstructured/sensorial**
  - actions.py
  - coreferences.py
  - entities.py
  - \_\_init\_\_.py
  - sentiment\_sam.py
  - utils.py
  - wrapper\_syntaxnet.py
- **leto/leto/unstructured/structural**
  - \_\_init\_\_.py
  - pack\_actions.py
  - pack\_entities.py
- **leto/leto-ui**
  - app.py
  - \_\_init\_\_.py
  - static
  - templates
- **leto/leto-ui/static/css**
  - bootstrap.min.css
  - font-awesome.css
  - font-awesome.min.css
  - style.css
- **leto/leto-ui/static/fonts**
  - FontAwesome.otf
  - fontawesome-webfont.eot
  - fontawesome-webfont.svg
  - fontawesome-webfont.ttf
  - fontawesome-webfont.woff
  - fontawesome-webfont.woff  
2
- **leto/leto-ui/static/js**
  - action\_trainer.js
  - bootstrap.bundle.min.js
  - index.js
  - jquery.min.js
  - lodash.js
  - ontology.js
  - vue.js
- **leto/leto-ui/templates**
  - index.html
  - ontology.html
- **leto/pipelines**
  - 1\_sensors.py
  - 2\_structures.py
  - 3\_knowledge.py
  - 4\_mapping.py
  - 5\_tagging.py
  - 6\_evaluators.py
  - 7\_generators.py
  - 9\_debug.py
  - \_action\_ontology.py
  - api.py
  - \_\_init\_\_.py
  - tasks
  - \_utils.py

**Requerimientos/Requirements:** Docker 20.10 o superior instalado, Docker-Compose 1.27 o superior instalado. 8 GB de RAM o superior, 4 GB de disco duro libre en sistema.

**Instalación/Installing:** Descargar el código fuente del repositorio de control de

versiones. En principio se puede usar en cualquier sistema operativo, pero sólo ha sido probado en Ubuntu Linux.

**Ejecución/Run:** Ejecutar el script ``leto.sh prod`` en la raíz del código fuente, o en su defecto, utilizar docker-compose de la manera convencional para ejecutar el proyecto de forma aislada en un contenedor de Docker.

Una vez en ejecución, navegar a ``localhost:3000`` para acceder a la interfaz web.

### **Dependencias/Dependencies:**

El listado completo de dependencias se encuentra en el archivo requirements.txt. Aquí se incluye un listado de las dependencias que se descargarán automáticamente al construir la imagen de Docker. No es necesario instalar explícitamente estas dependencias. Todas las dependencias son módulos o bibliotecas en el lenguaje de programación Python:

- flask (BSD-3-Clause)
- flask-restful (BSD-3-Clause)
- graphviz (MIT)
- neuralcoref (MIT)
- nltk (Apache 2.0)
- owlready (LGPL 3.0)
- owlready2 (LGPL 3.0)
- psutil (BSD-3-Clause)
- pytest (MIT)
- rdflib (BSD-3-Clause)
- spacy (MIT)
- textblob (MIT)
- wikipedia (MIT)

### **Referencias/References**

[EMA2019] Estevez-Velarde, S., Montoyo, A., Almeida-Cruz, Y., Gutiérrez, Y., Piad-Morffis, A., & Muñoz, R. (2019, September). Demo Application for LETO: Learning Engine Through Ontologies. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)* (pp. 276-284).

[EGM2018] Estevez-Velarde, S., Gutierrez, Y., Montoyo, A., Piad-Morffis, A., Munoz, R., & Almeida-Cruz, Y. (2018). Gathering object interactions as semantic knowledge. In *Proceedings on the International Conference on Artificial Intelligence (ICAI)* (pp. 363-369). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).